ไวรัสบนแผ่นดิสค์ ตอน ชำแหละไวรัส

ชาญชัย สัจจอิสริยวุฒิ

สุรเมธ สัจจอิสริยวุฒิ

จากบทความของผู้เขียนตอนที่แล้วได้กล่าวถึงการตรวจหา รักษา และแก้ไข แผ่นดิสค์ที่ติดเชื้อของไวรัสในที่นี้เป็นไวรัสที่เรียกว่า Virus(c) Brain เนื่องจากในปัจจุบันพบว่ามีไวรัสหลายตัวที่แพร่อยู่ในประเทศไทย และยังมีอีกหลายตัวที่เขียนขึ้นโดยคนไทยเราเอง

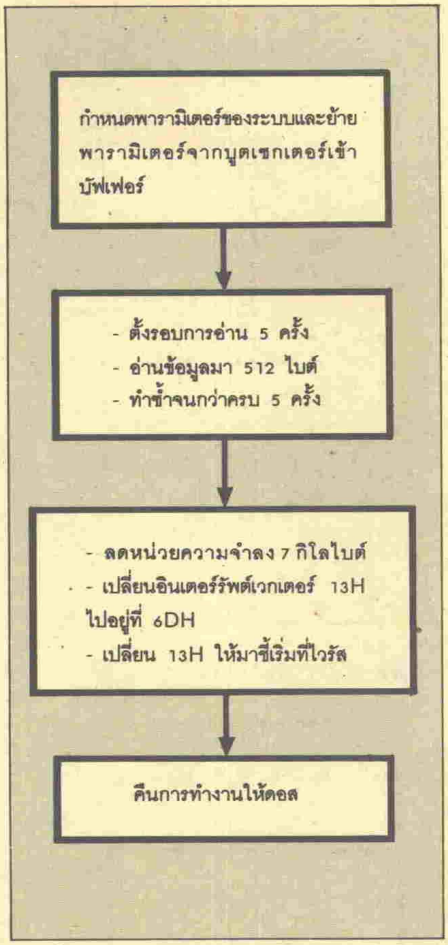
สาเหตุสำคัญที่ผู้เขียนเขียนบทความตอนนี้เพิ่มเติม เนื่องจากต้องการเตือนถึงผู้ที่มีความรู้ความสามารถในการเขียนโปรแกรมแต่มีความคิดที่มีเจตนาหรือไม่เจตนาที่จะคอยแกล้งผู้อื่นโปรดระงับการกระทำของท่าน ซึ่งท่านได้เพียงความสะใจเท่านั้น แต่จะทำให้ความก้าวหน้าทางเทคโนโลยีของประเทศทางคอมพิวเตอร์และทางเศรษฐกิจเกิดความเสียหายเป็นอย่างมาก ลองคิดดูว่าถ้าไวรัสของท่านเกิดแพร่เข้าไปในหน่วยงานของทางราชการไม่ว่าหน่วยงานใดจะทำให้เกิดผลเช่นใดกับประเทศ ผู้เขียนขอวิงวอนให้ท่านหยุดการกระทำของท่านเสีย หรือถ้าทำไปแล้วกรุณาเขียนโปรแกรมแก้ และเผยแพร่ให้แก่ส่วนรวม

เพื่อให้ผู้ใช้ได้ตระหนักถึงอันตรายอันเกิดจากไวรัส และเพิ่มความเข้าใจในขั้นตอนการทำงานของตัวโปรแกรมไวรัส ผู้เขียนจะขอกล่าวถึงวิธีการและขั้นตอนในการทำงานของตัวโปรแกรมไวรัสโดยสังเขป ดังนั้น ในบทความตอนนี้ผู้เขียนขอเรียกว่าตอน "ชำแหละไวรัส"

เริ่มชำแหละ

เนื่องจากขั้นตอนในการทำงานของไวรัสเมื่อพิจารณาแล้วค่อนข้างซับซ้อนดังนั้นในการอธิบายจึงขออ้างอิงถึงในรูปของบล็อกไดอะแกรมดังรูปที่ 1,2 เริ่มต้นจะขอกล่าวเป็นลักษณะการทำงานโดยคร่าว ๆ ของตัวไวรัสก่อนเพื่อให้เกิดความเข้าใจมากขึ้นในขั้นตอนของการชำแหละ ซึ่งหลักการทำงานนี้เป็นหลักการที่ผู้เขียนได้ทำการค้นคว้าและประสพมา ถ้ามีท่านผู้อ่านท่านใดที่พบการทำงานอื่นใดที่นอกเหนือจากนี้ กรุณาบอกกล่าวให้ทราบโดยทั่วกันจะเป็นพระคุณอย่างยิ่ง

ไวรัสเป็นโปรแกรมที่คอยดักการทำงานที่เกี่ยวกับดิสค์ทุกกรณี ดังนั้นเมื่อใดที่ดอสทำการติดต่อกับดิสค์จะต้องผ่านตัวโปรแกรมไวรัสเสมอไม่ว่าขั้นตอนนั้นเป็นการเขียนหรือการอ่านข้อมูลจากดิสค์ก็ตาม การทำงานของไวรัสเริ่มต้นด้วย เมื่อเราบูตด้วยแผ่นดิสค์ที่มีไวรัสอยู่ไม่ว่าแผ่นนั้นจะมีดอสหรือไม่มีก็ตาม ตัวโปรแกรมไวรัสซึ่งอยู่ในบูตเซกเตอร์ (Cylinder 0, Head 0, Sector 1 ในกรณีของฟลอปปี้ดิสค์)



รูปที่ 1 แสดงบล็อกไดอะแกรมของการทำงานโปรแกรมไวรัสส่วนต้น

จะเข้าไปอยู่ในหน่วยความจำ (ซึ่งปกติเครื่องต้องทำงานตามโปรแกรมที่ได้จากส่วนบูตเซกเตอร์นี้เพื่อทำการอ่านข้อมูลจากดิสก์ต่อไป) สำหรับไวรัสก็เช่นกัน เครื่องจะทำงานตามโปรแกรมไวรัสส่วนต้นซึ่งอยู่บริเวณบูตเซกเตอร์นี้ จากนั้นตัวโปรแกรมส่วนต้นนี้ก็ทำการอ่านส่วนอื่นของโปรแกรมไวรัสมาไว้ในหน่วยความจำ จากนั้นจึงเรียกตัวดอสมาทำงานต่อไป ดังนั้นไวรัสจึงสามารถควบคุมการทำงานของดอสได้ทุกครั้งที่มีการติดต่อกับดิสก์ ถ้าเรานำแผ่นที่ไม่มีไวรัสมาใช้งานกับเครื่องที่มีไวรัสอยู่ในหน่วยความจำ ตัวไวรัสในเครื่องจะทำการตรวจสอบดูว่าแผ่นที่ใช้งานในขณะนี้มีตัวไวรัสอยู่หรือยัง ถ้ายังไม่มีไวรัสจะทำการย้ายบูตเซกเตอร์เดิมของแผ่นไปไว้ที่บริเวณที่ว่างตำแหน่งแรกของแผ่นดิสก์ (ถ้าไม่มีที่ว่างพอหรือไม่ว่าง ไวรัสจะข้ามการทำงานไปทำงานตามปกติโดยไม่มีการเขียนตัวไวรัสลงบนแผ่น) แล้วจัดการย้ายตัวไวรัสส่วนต้นจากหน่วยความจำในเครื่องลงที่บูตเซกเตอร์แทน และย้ายส่วนอื่น ๆ ลงไปไว้ต่อท้ายบูตเซกเตอร์เดิมที่ถูกย้ายไป จากนั้นโปรแกรมจะทำการกำหนดบริเวณที่ตัวไวรัสไปอยู่นี้ว่าเป็นส่วน bad sector ไปเป็นขนาด 3 คลัสเตอร์ (การกำหนดนี้จะกำหนดที่ FAT (File Allowcation Table) ของแผ่นดิสก์) ตำแหน่งของเซกเตอร์ที่ไวรัสไปกำหนดเป็น bad sector หาได้จาก

ขนาดบริเวณ bad sector
= ที่เก็บตัวไวรัส
+ บูตเซกเตอร์เดิมของดอส
= 3 คลัสเตอร์

ซึ่งการกำหนดให้เป็น bad sector ทำให้การใช้งานดิสก์อย่างปกติไม่อาจมาใช้งานส่วนนี้ได้ เนื่องจากปกติแล้วเมื่อดอสต้องการจะติดต่อกับส่วนใดบนแผ่นดิสก์ดอสจะมาตรวจสอบส่วน FAT ว่าส่วนที่ต้องการติดต่อนี้ ว่าง ไม่ว่าง, หรือเสียหรือไม่ ถ้าเสียดอสจะไม่มาทำการติดต่อกับส่วนนี้เพื่อเป็นการรักษาอายุการใช้งานของตัวอ่านไม่ให้มีการอ่านส่วนเสียซึ่งอาจเป็นส่วนเสียที่เกิดจากการขีดข่วน ซึ่งมีผิวของแผ่นไม่เรียบ หลังจากกำหนดให้เป็น bad sector แล้ว ไวรัสก็จะโอนการทำงานต่อให้ไบออสทำงานไปอย่างปกติ

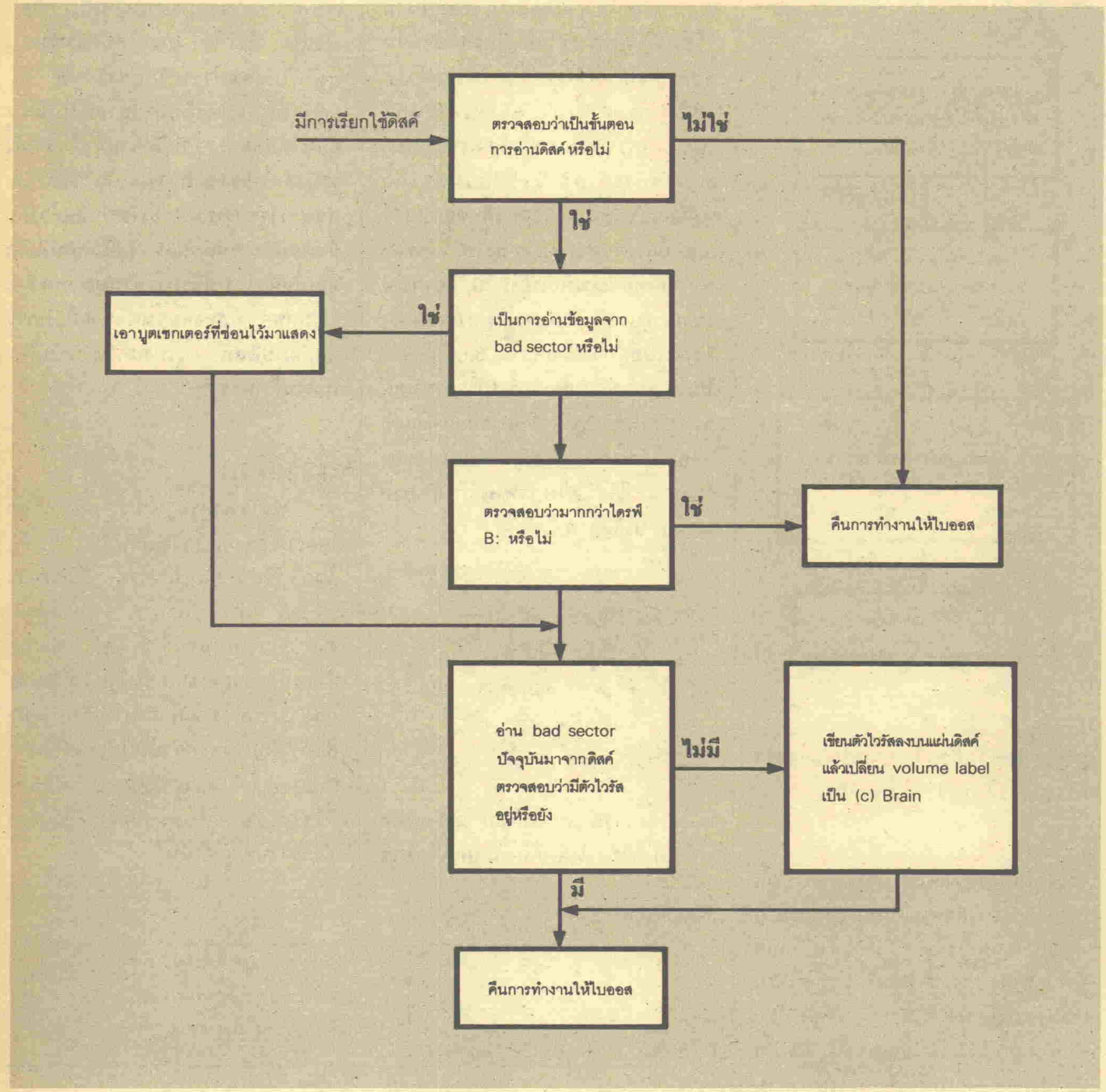
ขั้นตอนดังที่กล่าวมาแล้วเป็นขั้นตอนโดยคร่าว ๆ ที่เกิดขึ้นเมื่อไวรัสทำงาน ต่อไปเราจะพิจารณาตัวไวรัสทีละส่วน การทำงานของไวรัสปกติแล้วมี 2 ขั้นตอนคือ ขั้นตอนแรกคือขั้นตอนที่เกิดขึ้นเมื่อตัวไวรัสเข้ามาอยู่ในหน่วยความจำ ขั้นตอนที่สองคือขั้นตอนที่เมื่อมีการติดต่อกับดิสก์ ตัวโปรแกรมไวรัสก็จะทำงานอีกครั้ง

ส่วนแรก

พิจารณาตามรูปที่ 1 เป็นบล็อกไดอะแกรมของไวรัสส่วนต้น เริ่มต้นด้วยไวรัสจะกำหนดค่าของรีจิสเตอร์ SS,DS มีค่าเท่ากับ CS จากนั้นนำค่าจากบริเวณส่วนพารามิเตอร์ ซึ่งเก็บตำแหน่งของบูตเซกเตอร์เดิมเป็น track, side, sector มาแล้วย้ายไปเก็บบริเวณส่วนที่เป็นหน่วยความจำใช้งานชั่วคราว (buffer) ซึ่งตำแหน่งของพารามิเตอร์ที่ไวรัสใช้จะอยู่ในส่วนของบูตเซกเตอร์ ซึ่งเป็นไปดังต่อไปนี้

ตำแหน่งไบต์	หน้าที่
06	Side ของบูตเซกเตอร์เก่า
07	Sector ของบูตเซกเตอร์เก่า
08	Cylinder ของบูตเซกเตอร์เก่า

จากนั้นก็ทำการอ่านตัวโปรแกรมไวรัสส่วนที่เหลือเข้ามาโดยแบ่งการอ่านออกเป็น 5 ครั้ง ๆ ละ 512 ไบต์ รวมเป็น 2560 ไบต์ ถ้าการอ่านข้อมูลเกิดข้อผิดพลาดขึ้นตัวไวรัสจะพยายามอ่านซ้ำจนกว่าจะเกิดผิดพลาดครบ 4 ครั้งก็จะกระโดดเข้า BASIC ROM ทันที แต่ถ้าการอ่านไม่ผิดพลาดก็กำหนดตำแหน่งของหน่วยความจำส่วนที่เป็นที่



รูปที่ 2 แสดงบล็อกไดอะแกรมของการทำงานโปรแกรมไวรัสส่วนที่สอง

บอกขนาดของหน่วยความจำของเครื่องว่าเป็น 640K, 256K หรืออื่น ๆ ให้มีค่าลดลงจากเดิม 7 กิโลไบต์ ขั้นตอนต่อไปคือทำการเปลี่ยนค่าอินเตอร์รัพต์เวกเตอร์ในตารางอินเตอร์รัพต์ (ซึ่งอยู่บริเวณ 1K แรกของหน่วยความจำ) โดยที่ย้ายค่าเดิมของอินเตอร์รัพต์ 13H ซึ่งปกติเป็นอินเตอร์รัพต์ของไบออสที่ใช้ในการติดต่อกับดิสค์ให้ไปอยู่ตำแหน่งของอินเตอร์รัพต์ 6DH แล้วกำหนดค่าของอินเตอร์รัพต์เวกเตอร์ของอินเตอร์รัพต์ 13H ให้มาชี้ที่จุดเริ่มต้นของตัวโปรแกรมไวรัส แล้วกำหนดค่าพารามิเตอร์ไว้แล้วโอนการทำงานให้กับบูตเซกเตอร์เดิมของแผ่น ให้ทำงานไปตามปกติ

ส่วนสอง

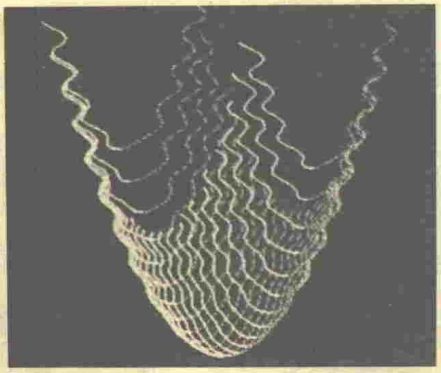
ส่วนสองนี้จะทำงานเมื่อผู้ใช้ทำการติดต่อกับดิสค์ ซึ่งปกติแล้วการติดต่อกับดิสค์ ต้องกระทำโดยผ่านไบออส

เพื่อให้สามารถใช้งานกับเครื่องที่คอม-แพตทิเบิ้ลได้ จากส่วนต้นไวรัสได้ทำการเปลี่ยนตำแหน่งของอินเตอร์รัพต์เวกเตอร์ 13H ให้มาผ่านตัวไวรัส ดังนั้นเมื่อมีการติดต่อกับดิสค์ เครื่องจะกระโดดมาทำงานที่ตัวโปรแกรมไวรัสเริ่มต้นจาก กำหนดให้อินเตอร์รัพต์ซ้อนได้ แล้วตรวจสอบว่าขั้นตอนที่เรียกใช้มาเป็นขั้นตอนอ่านหรือไม่ ถ้าไม่ใช่ก็จะส่งผ่านค่าให้ไบออสทำงาน จากนั้นก็ตรวจสอบว่าหมายเลขไดรฟ์ที่ต้องการเรียกใช้เกิน drive B: หรือไม่ ถ้าเกินให้ผ่านการทำงานให้ไบออส ขั้นต่อไปตรวจสอบว่าเป็นการอ่านข้อมูลจากบูตเซกเตอร์หรือไม่ ถ้าใช่ให้นำบูตเซกเตอร์เดิมที่ซ่อนไว้ขึ้นมาแสดง จากนั้นก็ทำการอ่านบูตเซกเตอร์ปัจจุบันขึ้นมาจากแผ่น ทำการตรวจสอบว่าเป็นแผ่นที่มีไวรัสอยู่แล้วหรือไม่ ถ้าเป็นให้ข้ามการทำงานทั้งหมดและโอนการทำงานให้ไบออส แต่ถ้ายังไม่มีไวรัสอยู่ตัวโปรแกรมไวรัสก็จะทำการเขียนข้อมูลลงบนแผ่นนั้น ๆ ก่อน แล้วจึงโอนการทำงานให้ไบออสต่อไป

จากขั้นตอนการทำงานทั้งหมดอาจจะมีข้อสงสัยบางอย่างในการทำงานของไบออสและดอส ให้อ่านเพิ่มเติมได้จาก

1. IBM Personal Computer Technical Reference
2. MS-DOS (version 1.0-3.2) Technical Reference Encyclopedia

เนื่องจากขั้นตอนการทำงานของไวรัสเป็นไปดังกล่าวมาแล้ว ดังนั้นแนวการแก้ไขของโปรแกรมศิริราชของผู้เขียนจึงแก้ตามสิ่งที่ไวรัสทำขึ้นซึ่งผู้เขียนพบเท่านั้น ท่านใดที่พบการทำงานที่มากกว่านี้กรุณาติดต่อผู้เขียนหรือผ่านทางวารสารไมโครฯ เพื่อผู้เขียนจะทำการแก้ไขโปรแกรมศิริราชเพิ่มเติมต่อไป



ขั้นตอนการทำงานของโปรแกรมศิริราชโดยคร่าว ๆ

เนื่องจากโปรแกรมไวรัส ทำการย้ายบูตเซกเตอร์เดิมของแผ่นไปซ่อนไว้ ดังนั้นแนวทางการแก้ไขคือการอ่านข้อมูลพารามิเตอร์จากบูตเซกเตอร์ปัจจุบันมาทำการคำนวณหาตำแหน่งของบูตเซกเตอร์เดิมที่ถูกซ่อนไว้ แล้วอ่านข้อมูลของส่วนนี้ขึ้นมา จากนั้นก็เขียนตัวบูตเซกเตอร์เดิมนี้กลับคืนที่เดิมคือ บริเวณ Cylinder 0, Head 0, Sector 1 ดังนั้นตัวโปรแกรมไวรัสก็จะหมดพิษสงไปทันที จากนั้นทำการคำนวณหาตำแหน่งที่ไวรัสไปซ่อนตัวอยู่ แล้วทำการแก้ FAT ให้เป็นใช้งานได้เหมือนเดิม

ขั้นตอนการทำงานที่สมบูรณ์

1. เมื่อโปรแกรมศิริราชเข้าสู่หน่วยความจำ จะเริ่มทำการค้นหาตัวไวรัสในหน่วยความจำทันที ถ้าเจอจะกำจัดตัวไวรัสออกจากหน่วยความจำของเครื่อง และแจ้งให้ทราบว่าตอนนี้ไวรัสถูกกำจัดออกไปแล้ว

2. ถามว่าต้องการตรวจหาไวรัสหรือไม่

3. ถ้าต้องการ เริ่มทำการตรวจหาไวรัสที่แผ่นดิสค์ ถ้าพบจะทำการเตือนว่ามีไวรัส และถามว่าต้องการกำจัดทิ้งหรือไม่ ถ้าไม่กำจัดออก โปรแกรมศิริราชจะเตือนว่ามีไวรัสอยู่ให้ระมัดระวังในการใช้งานดิสค์แผ่นนี้

4. ถ้ากำจัดออก โปรแกรมศิริราชจะรายงานถึงบริเวณตำแหน่งที่ไวรัสไปซ่อนตัวอยู่ และทำการแก้ไขดังวิธีที่ได้กล่าวมาแล้ว และทำการฉีดวัคซีนไม่ให้ไวรัสสามารถมาติดแผ่นดิสค์นี้ได้อีก

5. ทำการแก้ไข volume label ซึ่งถูกไวรัสเปลี่ยนไป จาก (c) Brain เป็น Siriraj100 เป็นการแจ้งให้ทราบว่าได้ทำการแก้ไขเรียบร้อยแล้ว

จุดประสงค์อีกประการหนึ่งในการเขียนบทความนี้เพื่อรายงานถึงคุณสมบัติเพิ่มเติมของตัวโปรแกรมไวรัส (c)Brain และทางผู้เขียนได้รับรายงานถึงจุดอ่อนของโปรแกรมศิริราช (Siriraj) ที่ผู้เขียนได้เขียนขึ้นดังนี้

1. โปรแกรมศิริราชไม่สามารถหาไวรัสพบได้ในบางครั้ง

2. โปรแกรมสามารถทำงานได้แต่หลังจากที่แก้ไขแล้ว ยังมีไวรัสอยู่

3. โปรแกรมทำการแก้ไขแผ่นแล้ว แต่ปรากฏว่าข้อมูลทั้งหมดกลับหายไป หรือ FAT เสีย

4. โปรแกรมตรวจพบไวรัส ในเครื่อง AT ทั้ง ๆ ที่ไม่มีไวรัส

5. หลังจากการใช้งานโปรแกรมแล้ว เมื่อใช้งานฟลอปปี้ดิสค์ปรากฏว่าทำงานผิดพลาด

ปัจจุบันทางผู้เขียนได้เขียนตัวโปรแกรมศิริราชออกมาหลายรุ่น ดังนี้

- Release 1.0

เป็นโปรแกรมศิริราชรุ่นแรกที่ออกมาใช้หน่วยความจำน้อย และทำงานได้ไม่ดีนัก

- Release 2.0

ได้เพิ่มเติมคุณสมบัติในการตรวจจับไวรัสในหน่วยความจำ และความสามารถในการกำจัดไวรัสที่อยู่ในหน่วยความจำออก

- Release 2.5 และ Release 2.5a

ได้เพิ่มเติมคุณสมบัติในการป้องกันการติดเชื้อไวรัส กล่าวอย่างง่าย ๆ คือฉีดวัคซีนกันไวรัสให้กับแผ่นดิสค์นั่นเอง

- Release 3.0

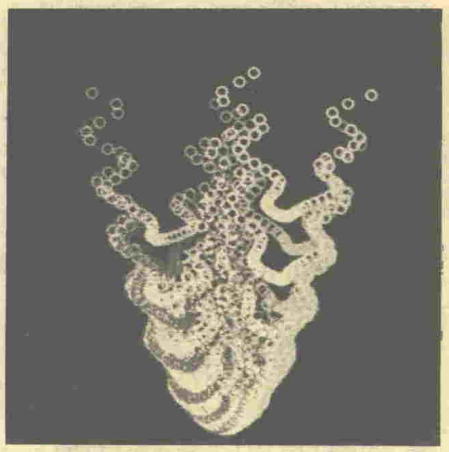
เป็นโปรแกรมศิริราชรุ่นใหม่ที่สุด ได้เพิ่มความสามารถในการแสดงบูต-เรคอร์ด เพื่อให้ผู้ใช้สามารถสังเกตหาไวรัสตัวใหม่อื่น ๆ ได้ง่ายขึ้น เพิ่มเติมความสามารถให้ทำการป้องกันการติดเชื้อไวรัสหลังจากที่ทำการรักษาแผ่นเรียบร้อยแล้ว และเพิ่มเติมความสามารถด้านจัดการจอภาพไม่ให้จอภาพเลอะ นอกจากนี้ได้ทำการแก้ไขข้อผิดพลาดอื่น ๆ ซึ่งเกิดขึ้นในรุ่นก่อน ๆ ทั้งหมด ทำให้สามารถใช้งานกับเครื่องรุ่น AT ขึ้นไปได้อย่างสมบูรณ์

สำหรับปัญหาที่เกิดขึ้นในการใช้งานโปรแกรม ทางผู้เขียนจะขออธิบายปัญหาที่เกิดขึ้นของแต่ละหัวข้อดังที่กล่าวมาแล้วโดยคร่าว ๆ ดังนี้

1. โปรแกรมศิริราชไม่สามารถหาไวรัสพบได้ในบางครั้ง

- เป็นผลที่เกิดขึ้นในการใช้งานในโปรแกรมรุ่นแรก (release 1.0) เนื่องจากโปรแกรมรุ่นแรกนี้ ผู้เขียนได้เร่งรีบเขียนออกมา จึงยังตรวจความสามารถของโปรแกรมไวรัสได้ไม่ครบ ซึ่งความสามารถอันหนึ่งของไวรัสคือ เมื่อใดที่โปรแกรมใด ๆ ก็ตามเรียกมาอ่านบูต-เซกเตอร์ของแผ่นที่มีไวรัส (ไวรัสต้องอยู่ในหน่วยความจำของเครื่องแล้ว) ไวรัสจะจัดการเปลี่ยนให้ไปอ่านบูต-เซกเตอร์ของเดิมที่มันเก็บซ่อนเอาไว้ ทำให้โปรแกรมศิริราชรุ่นแรก ซึ่งทำงานโดยผ่านดอสถูกโปรแกรมไวรัสหลอกได้ วิธีแก้ปัญหาเมื่อใช้โปรแกรมรุ่นนี้คือบูตดอสที่แน่ใจว่าไม่มีไวรัสอยู่ โปรแกรมจะทำงานได้ตามปกติ

2. โปรแกรมสามารถทำงานได้



แต่หลังจากที่แก้ไขแล้ว ยังมีไวรัสอยู่

- ปัญหาในหัวข้อนี้เกิดขึ้นเนื่องจากกรณีเดียวกับในหัวข้อที่ 1

3. โปรแกรมทำการแก้ไขแผ่นแล้ว แต่ปรากฏว่าข้อมูลทั้งหมดกลับหายไป หรือ FAT เสีย

- สำหรับในกรณีนี้ ผู้เขียนเองคิดว่าไม่น่าเป็นไปได้เนื่องจากโปรแกรมทำงานผ่านดอสจึงไม่น่าจะเกิดปัญหานี้ขึ้น หลังจากทำการตรวจสอบพบว่าตัวเครื่องฟลอปปี้ดิสค์ที่ใช้ทำงานได้ไม่ดี ทำให้การเขียนข้อมูลเกิดเพี้ยนไปได้

4. โปรแกรมตรวจพบไวรัส ในเครื่อง AT ทั้ง ๆ ที่ไม่มีไวรัส

- เป็นปัญหาที่เกิดขึ้นกับโปรแกรมรุ่น 2.0 ซึ่งได้เพิ่มความสามารถในการตรวจหาตัวโปรแกรมไวรัสในหน่วยความจำ ดังนั้นการใช้งานโปรแกรมรุ่นนี้ให้ดูว่าหน่วยความจำเครื่องเกิดผิดพลาดไปหรือไม่ ถ้าหน่วยความจำลดลงเช่น จากเครื่อง 640 กิโลไบต์ เหลือเพียง 633 กิโลไบต์ ให้แน่ใจได้ว่ามีไวรัสอยู่ในเครื่องให้ทำการบูตเครื่องใหม่ ปัจจุบันข้อผิดพลาดนี้ได้ทำการแก้ไขเรียบร้อยแล้วในโปรแกรมรุ่น 3.0

5. หลังจากการใช้งานโปรแกรมแล้ว เมื่อใช้งานฟลอปปี้ดิสค์ปรากฏว่าทำงานผิดพลาด

- เป็นความผิดพลาดของผู้เขียนเนื่องจากเข้าไปดัดแปลงการส่งข้อมูลให้แก่ดอสและคืนค่าต่าง ๆ ให้ไม่ครบทำให้ดอสทำงานผิดพลาดไปจากเดิม และทางผู้เขียนได้ตรวจพบ และทำการแก้ไขเรียบร้อยแล้วในโปรแกรมรุ่น 3.0

เนื่องจากในโปรแกรมศิริราชตั้งแต่รุ่น 2.0 ขึ้นมามีความสามารถในการตรวจหาไวรัสในหน่วยความจำ และในรุ่น 3.0 ได้เพิ่มความสามารถในการป้องกันการติดไวรัสด้วย แต่เนื่องจากวิธีการตรวจและป้องกันมีข้อยุ่งยากพอควร ผู้เขียนจึงขอละไว้เป็นบทความตอนหน้าซึ่งเป็นการชำแหละไวรัสอย่างละเอียดแบบไบต์ต่อไบต์

คุณสมบัติที่เพิ่มเติมของไวรัสคือ ไม่สามารถติดกับฮาร์ดดิสค์ได้ แต่จากที่ผู้เขียนได้ทำการทดสอบพบว่าถึงแม้ว่าไวรัส (c) Brain นี้จะไม่ติดฮาร์ดดิสค์ก็ตาม แต่จะทำให้พื้นที่ว่างของหน่วยความจำของฮาร์ดดิสค์ลดลงประมาณ 1 กิโลไบต์ และไม่สามารถเรียกคืนมาได้ ไม่ว่าวิธีการใด ๆ แม้แต่การใช้คำสั่ง Format เพื่อทำการ Format ใหม่ แต่สามารถเรียกคืนได้วิธีการเดียวคือ ทำการ format hard disk ใหม่เหมือนกับว่าเป็น hard disk ที่ยังไม่ได้มีการใช้งานมาก่อนคือ format เพื่อจัด cylinder ก่อน แล้วสั่ง fdisk เพื่อทำการแบ่ง partition แล้วจึงทำการ format ตามปกติต่อไป ขั้นตอนทั้งหมดนี้ค่อนข้างยุ่งยากอาจให้ทางผู้จัดจำหน่ายเครื่องคอมพิวเตอร์จัดการให้จะได้แน่ใจว่าไม่ผิดขั้นตอน

สำหรับตัวโปรแกรมศิริราช Release 1.0, 2.0, 2.5, 2.5a สามารถหาก๊อบปี้ได้ทั่วไปตามร้านจัดจำหน่ายเครื่องคอมพิวเตอร์ และทางผู้เขียนขอวิงวอนทางร้านให้จำหน่ายโปรแกรมนี้ซึ่งผู้เขียนเขียนขึ้นเพื่อให้เป็นประโยชน์แก่ส่วนรวมโดยไม่หวังสิ่งตอบแทนใด ๆ

ขั้นตอนการใช้งานโปรแกรม ศิริราช ในการตรวจหาไวรัสและทำการรักษา

```
/ S / I / R / I / R / A / J / .... Release 3.0
Program to diagnose and treat viral in memory & diskette.
By  Ext.Chanchai sajjaisariyavut.  Siriraj' 94.
    Suramet  sajjaisariyavut.  Applied Physics. KMITL
Congratulations! You are using the best real protection program against virus.

CHECK SYSTEM.
MEMMORY SPACE = 640 Kbyte.
GOOD! NO VIRUS IN MEMORY.

Do you want to Search for Virus in drive A: ? y

================================ BOOT SECTER ================================
J.4...........Welcome to the  Dungeon          (c) 1986 Brain.& Amjads (pvt) Lt
d   VIRUS_SHOE  RECORD   v9.0   Dedicated to the dynamic memories of millions of
virus who are no longer with us today - Thanks GOODNESS!!       BEWARE OF THE e
r..VIRUS  : \this program is catching      program follows after these messeges.
.... $#@%$@!! ←ห→ู→เพ.ื่ .¦ข.¦■..¦┬..¦ฟ.ป..ฝ.~*.K.┐ล..ก..-..ค..ณ.ใ→ยภ.¦ม..
ป..ฆ.บ..PฮQSป..Q┼6.¦ด.■..¦บ..ภ.s.ถ.ภ.Yภ.Y[Yล .¦ยข.¦<.u.ษ..¦. .¦ยข.¦<.u.ษ..
¦...¦ล....2#MYก└พล..~.ภ!ข<_..
==============================================================================

Dangerous! Detected virus in drive A:!
Do you want to KILL virus in drive A: ? (Y/N)n
Warning...Virus in drive A:.
Do you want to KILL virus in drive A: ? (Y/N)y
 infected organ.
 T13 Sec1 side0
Ready! Now diskette in drive A: has a good health.
OK! Now the disk in drive A is protected from virus for ever.
```

รูปที่ 3 ขั้นตอนการใช้งานโปรแกรมศิริราชในการตรวจหาไวรัสและทำการรักษา

ขั้นตอนในการป้องกันไม่ให้ ไวรัส มาติดแผ่นดิสค์ โดยใช้โปรแกรมศิริราช

```
Do you want to Search for Virus in drive A: ? n
Select disk drive ( 'P' to Protect, 'Q' to Quit ) : p

Please insert the disk which you want to Protect from virus in drive A:
And Press ENTER to continue
OK! Now the disk in drive A is protected from virus for ever.

Do you want to Search for Virus in drive A: ? n
Select disk drive ( 'P' to Protect, 'Q' to Quit ) : q

O.K. QUIT.


          THANKS TO

                (1). SIRIRAJ HOSPITAL. Bangkok. THAILAND.
                (2). K.M.I.T. Ladkrabang Bangkok. THAILAND.
                (3). L.C. Hard & Software COMPUTER. THAILAND.
```

รูปที่ 4 ขั้นตอนในการป้องกันไม่ให้ไวรัสมาติดแผ่นดิสค์โดยใช้โปรแกรมศิริราช

ทั้งสิ้น ดังนั้นเพื่อส่วนรวมกรุณาก๊อปปี้โปรแกรมนี้โดยไม่คิดค่าใช้จ่ายหรือถ้าคิดกรุณาคิดเพียงเล็กน้อย (ประมาณ 5-10 บาท ผู้เขียนคิดว่าเพียงพอสำหรับค่าไฟฟ้ากับค่าเสียเวลาในการก๊อปปี้ซึ่งเสียเวลาเพียง 1-2 นาทีเท่านั้น) ทางผู้เขียนเองขอสงวนสิทธิ์อันพึงมีในโปรแกรมนี้ตามพระราชบัญญัติกฎหมายลิขสิทธิ์ พุทธศักราช 2521 อนุญาตให้ทำซ้ำเพื่อการใช้งานได้ โดยคิดค่าใช้จ่ายแต่ไม่ใช่เพื่อการหากำไร

สำหรับผู้ที่สนใจในโปรแกรมศิริราช Release 3.0 สามารถก๊อปปี้ได้ที่

- ร้าน ไมไครซ๊อฟ ซอยมังกร 1
- บริษัท ซุปพีเรียร์คอมพิวเตอร์ จำกัด ชั้น 2 โรงภาพยนตร์ลิโด้
- บริษัท IRC จำกัด ชั้น 2 โรงภาพยนตร์ลิโด้

สำหรับผู้ที่สนใจในตัว source code ของ virus หรือ มีปัญหาในการใช้งานโปรแกรมศิริราช หรือพบข้อบกพร่องของโปรแกรม ศิริราช หรือมีข้อคิดเห็นต่าง ๆ ทางผู้เขียนยินดีตอบรับและรับฟังปัญหาและข้อผิดพลาดที่เกิดขึ้นด้วยความยินดีเป็นอย่างยิ่ง กรุณาติดต่อกับผู้เขียนได้ที่

สุรเมธ สัจจอิสริยวุฒิ
ภาควิชาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง
ถนนฉลองกรุง ลาดกระบัง กรุงเทพฯ 10520

ท้ายนี้ผู้เขียนขอขอบคุณร้านไมโครซ๊อฟ และ บริษัทซุปพีเรียร์คอมพิวเตอร์ที่กรุณาให้ผู้เขียนทำการทดสอบโปรแกรมศิริราชบนเครื่องคอมพิวเตอร์ทุกรุ่น ตั้งแต่รุ่น XT, AT และเครื่อง 386

หมายเหตุ ถ้าผู้อ่านท่านใด คีย์โปรแกรมแล้วคอมพายล์ไม่ได้หรือมีปัญหาใด ๆ กรุณาติดต่อกับผู้เขียนโดยตรงตามที่อยู่ให้ไว้

รูปที่ 5 แสดง sort code ของโปรแกรม SIRIRAJ R.3.0

```
 *
  PROGRAM..............  SIRIRAJ
  Release..............  3.0
  Perpouse.............  to treat the virus diskette and protect diskette
                         from virus
  Language.............  Turbo-C Version 1.5
  Last update..........  August 5, 1988
  Time.................  01:57:00
  Programmer...........  LOUIS & LEK
  Copyright............  (c) August 1988
  Note.................  This program is PUBLIC DOMAIN not for sale
*/
#include <dos.h>
#include <ctype.h>
#include <conio.h>
extern void vimem();
struct text_info txtinf;
main()
{ unsigned char data[1024];
  int a,sector,b,c,drive,drv;
  clrscr();
  drive = 0;
  b = 1;
  gettextinfo(&txtinf);
  highvideo();
  textattr(LIGHTCYAN);
  printf ("/ S / I / R / I / R / A / J / .... Release 3.0\n"
          "Program to diagnose and treat viral in memory & diskette.\n"
          "By  Ext.Chanchai sajjaisariyavut.  Siriraj' 94.\n"
          "    Suramet  sajjaisariyavut.  Applied Physics. KMITL\n");
  cprintf("\nCongratulations! You are using the best real protection program against virus.");
  normvideo();
  cprintf("\n\n");
  textattr(LIGHTRED+(GREEN<<4)+BLINK);
  cprintf("CHECK SYSTEM.\n");
  textattr(txtinf.normattr);
  sound(75);
  delay(3000);
  nosound();
  for ( a = 500;a < 5000; a += 50)
      {
       sound(a);
       delay(50);
      }
  nosound();
  vimem();
  window(1,wherey(),80,25);
  delay(1000);
ASK:
  cprintf("Do you want to Search for Virus in drive %c: ? ",drive+65);
  sound(1000);
```

```
delay(100);
nosound();
while ( (a=toupper(getche())) == 'Y' )
{
  drv = drive;
  a = absread(drive,1,0,data);
  if ( a == -1 )
      cprintf("\nDrive does not Ready! Please check your drive %c: ",drive+65);
  while ( a == -1 )
     {
      sound(25);
      getch();
      nosound();
      a = absread(drive,1,0,data);
     }
  cprintf("\n\n================================ BOOT SECTER ================================");
  for ( b = 0; b<512; b++)
     {
      a = data[b];
      if ( data[b]<32 )
         a = 0x2e;
      cprintf ("%c",a);
     }
  cprintf("\n============================================================================");
  if ((data[0] == 0xfa) && (data[1] == 0xe9) && (data[2] == 0x4a) && (data[3] == 0x01) && (data[4] == 0x34)
  && (data[5] == 0x12))
   {
     drive = drv;
     textattr(LIGHTCYAN+(RED<<4)+BLINK);
     cprintf("\nDangerous!");
     textattr(txtinf.normattr);
     cprintf(" Detected virus in drive %c:!\n",drive+65);
     sound(300);
      delay(150);
     sound(500);
     delay(150);
     sound(250);
     delay(200);
     nosound();
     cprintf("Do you want to KILL virus in drive %c: ? (Y/N)",drive+65);
     if ((a=toupper(getche())) == 'Y')
        {
         cprintf("\nWait a few second... Infected Disk is still under treatment.\n");
         sound(75);
         sector = (data[8]*18)+data[7]+(data[6]*9)-1;
         cprintf("\n infected organ.");
         cprintf("\n Track%d Sector%d Side%d",data[8],data[7],data[6]);
         a = absread(drive,1,sector,data);
         data[04] = 0x34;
         data[05] = 0x12;
         abswrite(drive,1,0,data);
         absread(drive,2,1,data);
         a =(3*(sector-12)/4)+3;
         b= sector%4;
         if( b == 0)
```

```
            data[a+4] = data[a+4]&0xf0;
          else
            {
             data[a] = data[a]&0x0f;
             a++;
            }
          for( b=0;b<4;b++)
             data[a+b] = 0;
          abswrite(drive,2,1,data);
          abswrite(drive,2,3,data);
          a= b = 5;
          while ( data[a] != 8 && b<11)
              {
                absread(drive,1,b,data);
                for( c=0;c<496 ;c++)
                   {
                      if ((data[c] == 8) && (data[c-2] ==0x6e) && (data[c-5] ==0x72))
                         a = c;
                   }
                b++;
              }
          if (b<11)
             {
              b--;
              data[a-1] = data[a-2] = 0x30;
              data[a-9] = data[a-7] = 0x72;
              data[a-10] = data[a-8] =0x69;
              data[a-11] =0x53;
              data[a-6] = 0x61;
              data[a-5] = 0x6a;
              data[a-4] = 0x20;
              data[a-3] = 0x31;
              abswrite(drive,1,b,data);
             }
          cprintf("\nReady! Now diskette in drive %c: has a good health.\n",drive+65);
          cprintf("And diskette is protected from virus forever.\n");
          nosound();
        }
       else
        {
         textattr(LIGHTMAGENTA+(BLUE<<4)+BLINK);
         cprintf ("\nWarning...");
         textattr(txtinf.normattr);
         cprintf("Virus in drive %c:.",drive+65);
        }
    }
  else
   cprintf("\nDiskette in drive %c: is quite healthy.\n",drive+65);
  b = getch();
  clrscr();
  cprintf("Do you want to Search for Virus in drive %c: ? ",drive + 65);
  sound(1000);
  delay(100);
  nosound();
}
```

```
    cprintf("\nSelect disk drive ( 'P' to Protect, 'Q' to Quit ) : ");
    drv = drive;
    drive = toupper(getche());
    if ( drive == 'P')
        { drive = drv;
          emu(drv);
          clrscr();
          goto ASK;
        }
    if ( drive == 'Q')
        {
          clrscr();
          cprintf("O.K. QUIT.\n\n\n");
          textattr(YELLOW+(BLUE<<4)+BLINK);
          cprintf("          THANKS TO          \n");
          textattr(YELLOW);
          cprintf("\n                          (1). SIRIRAJ  HOSPITAL.  Bangkok.  THAILAND.");
          cprintf("\n                          (2). K.M.I.T. Ladkrabang Bangkok.  THAILAND.");
          cprintf("\n                          (3). L.C. Hard & Software COMPUTER. THAILAND.  \n");
          cprintf  ("                          ");
          textattr(LIGHTBLUE);
          cprintf("                                                \n\n\n");
          for(a = 5000; a >500; a -= 50)
             {
              sound(a);
              delay(50);
             }
          nosound();
          exit();
        }
    else
        {
          drive -= 65;
          if ( drive >3 || drive <0)
              drive = drv = 0;
          clrscr();
          goto ASK;
        }
}
void vimem()
{ int unpro[5] = { 0xcd, 0x6d, 0xca, 0x02, 0x00};
 unsigned long int a,b,c,dd,ee,ff,cc;
 unsigned int d,e,i;
 a  = peek(000,0x1b4);
 dd = peek(000,0x1b6);
 ee = peek(000,0x1b8);
 ff = peek(000,0x1ba);
 /*check memory */
 b = peek(000,0x413);
 c = b*1024;
 cc = b % 2;
 cprintf("MEMMORY SPACE = %d Kbyte.\n",b);
 if ( ((a != ee) || (dd != ff)) && cc == 1 )
    {
```

```
        poke(000,0x413,b+7);
        cprintf("Detected Virus in Your computer!\n");
        sound(300);
         delay(150);
         sound(500);
         delay(150);
         sound(250);
         delay(200);
         nosound();
        cprintf("Memory address of Virus.\n");
        d = c >> 4;
        e = c & 0xf;
        cprintf("SEGMENT = %X . OFFSET = %4X .\n",d,e);
        /* Clear virus in memory */
        for ( i=0;i <5; i++)
             pokeb (d, e+0x277+i ,unpro[i]);
        textattr(LIGHTMAGENTA + BLINK);
        cprintf("Virus in Memory to be inactive.\n");
        textattr(txtinf.normattr);
      }
    else
      cprintf("GOOD! NO VIRUS IN MEMORY.\n");
    cprintf("\n");
}

emu(int drive)
{
 int a,b;
 unsigned char dat[1024];
 cprintf("\n\nPlease insert the disk which you want to Protect from virus in drive %c:",drive+65);
 cprintf("\nAnd Press ENTER to continue");
 a = getch();
 b = drive;
 a = absread(drive,1,0,dat);
 drive = b;
 if ( a == -1 )
     cprintf("\nDrive does not Ready! Please check your drive %c:\n",drive+65);
 while ( a == -1 )
  {
   sound(25);
   getch();
   nosound();
   drive = b;
   a = absread(drive,1,0,dat);
  }
 dat[04]=0x34;
 dat[05]=0x12;
 dat[06]=00;
 dat[07]= 1;
 dat[08]=00;
 abswrite(drive,1,0,dat);
 cprintf("\nOK! Now the disk in drive %c is protected from virus for ever. n",drive+65);
 a = getch();
}
```